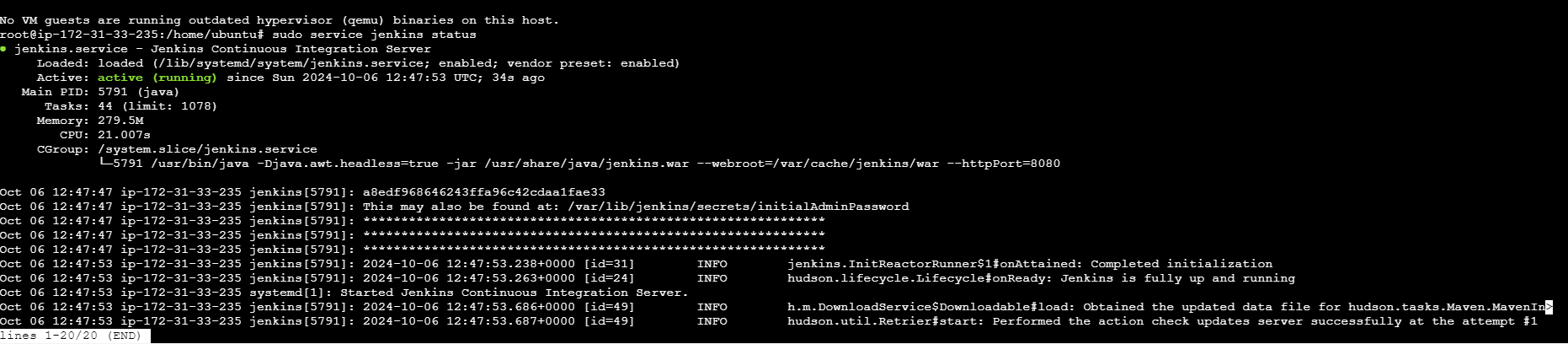
1. Create one GCP Ubuntu 22.04 instance and run below commands one by one. Make sure each command executed successfully before running next command:

sudo apt update

sudo apt install openjdk-11-jre

[https://pkg.jenkins.io/debian-stable binary/](https://pkg.jenkins.io/debian-stable%20binary/)

https://pkg.jenkins.io/debian binary/



**Exercise 2: Complete below tasks as part of this exercise:\*\***

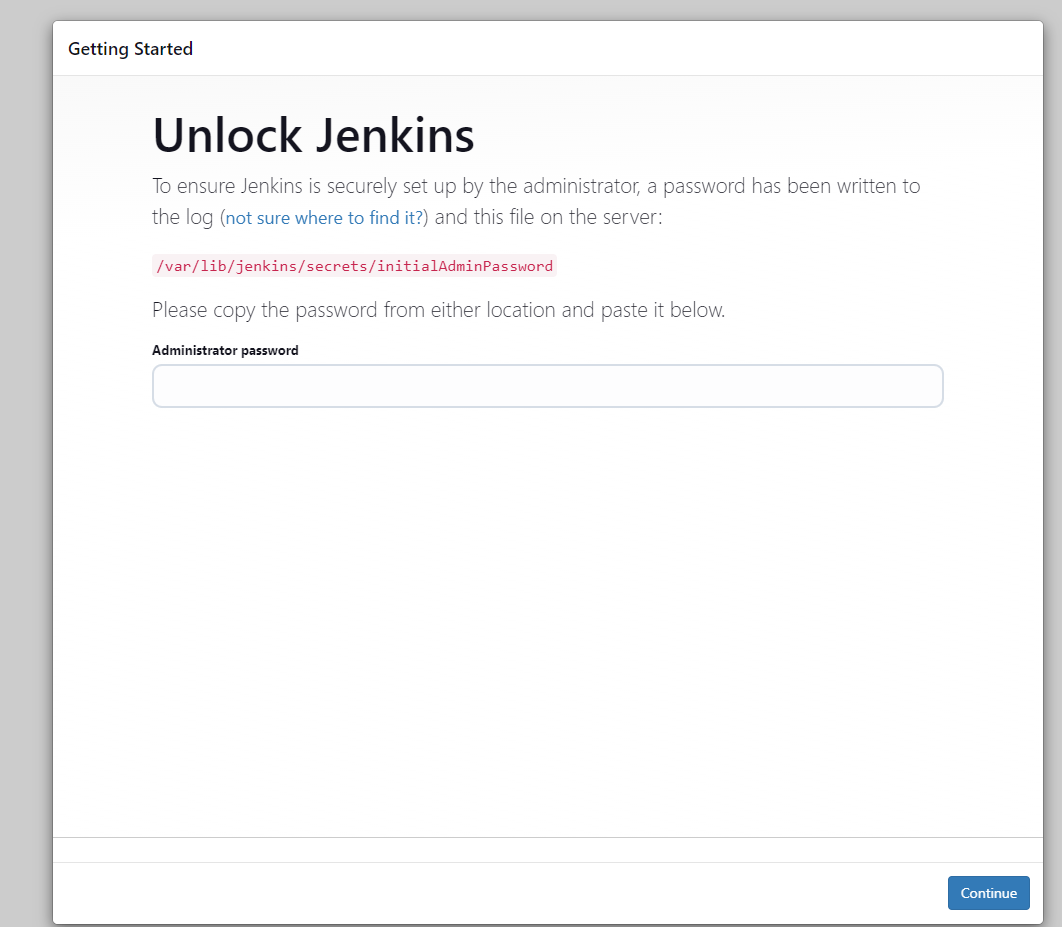
1. Create one more GCP server and configure it as slave node to Jenkins you installed in exercise 1:

***Solution:***

***a) Configure Jenkins***

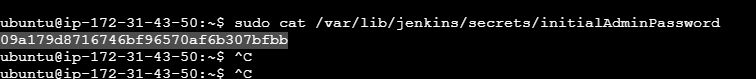
i. Open browser and enter masterIP:8080.

ii. You should land on a page like this:



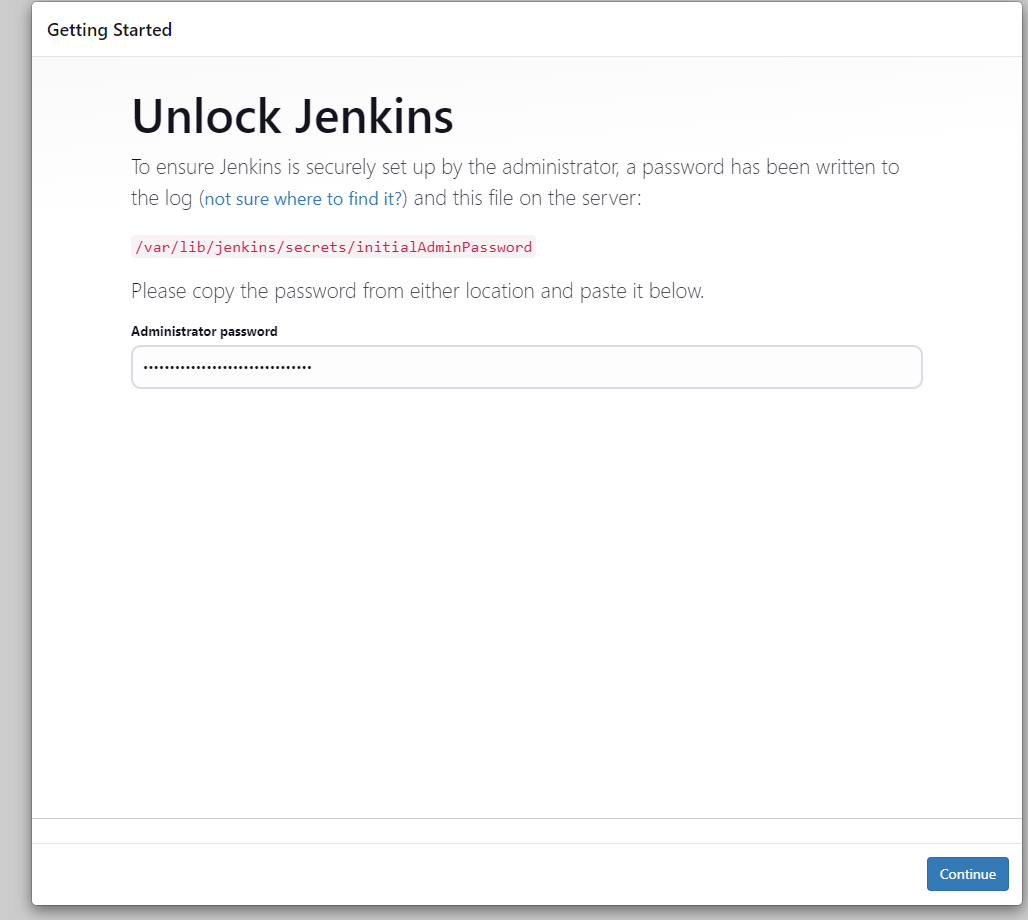
iii. Copy the path mentioned in the page and perform cat operation in master terminal.

sudo cat 09a179d8716746bf96570af6b307bfbb

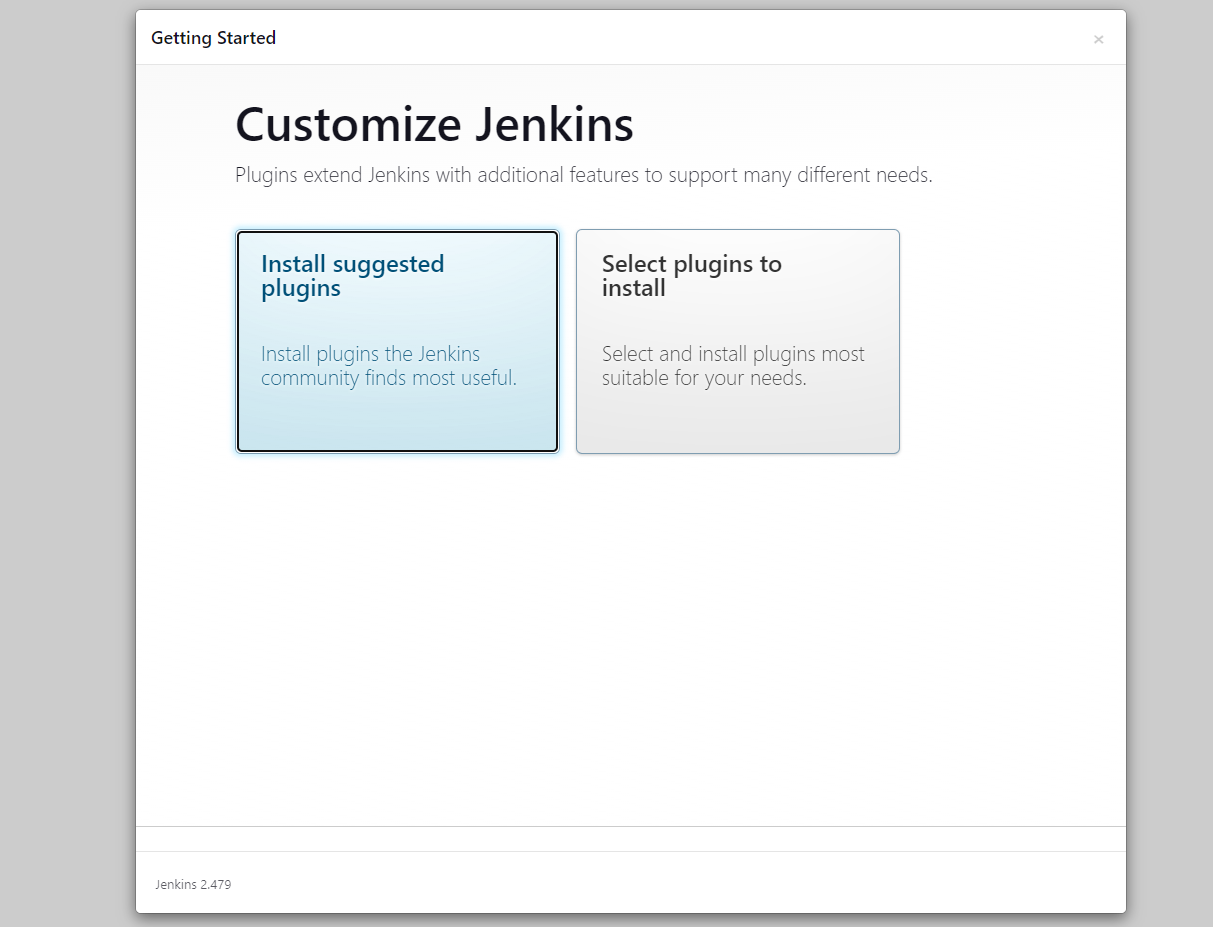


This will give us the password which we will use to unlock our Jenkins.

Copy the password from there and paste it on the Jenkins Server page.

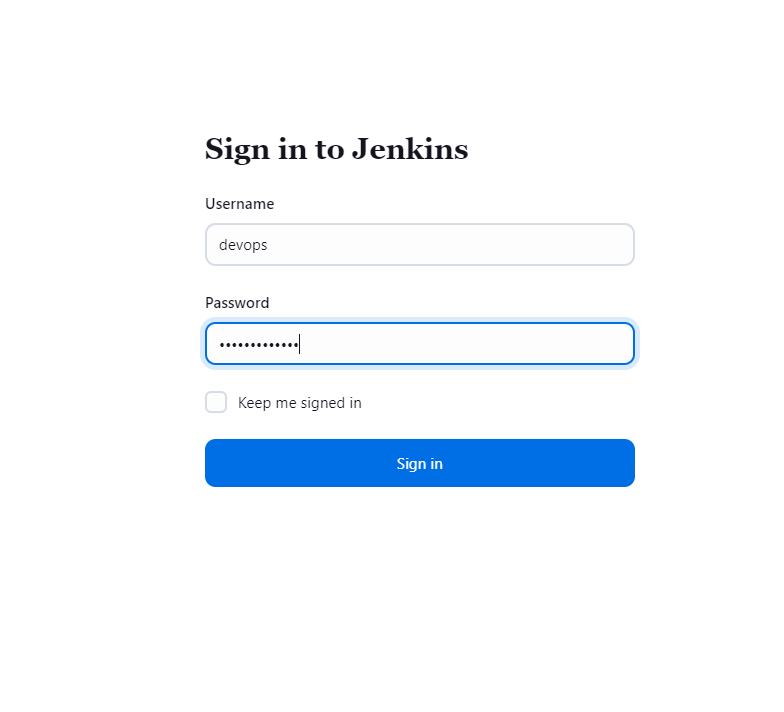


iv. Now click on continue. Then click on Install Suggested Plugins.

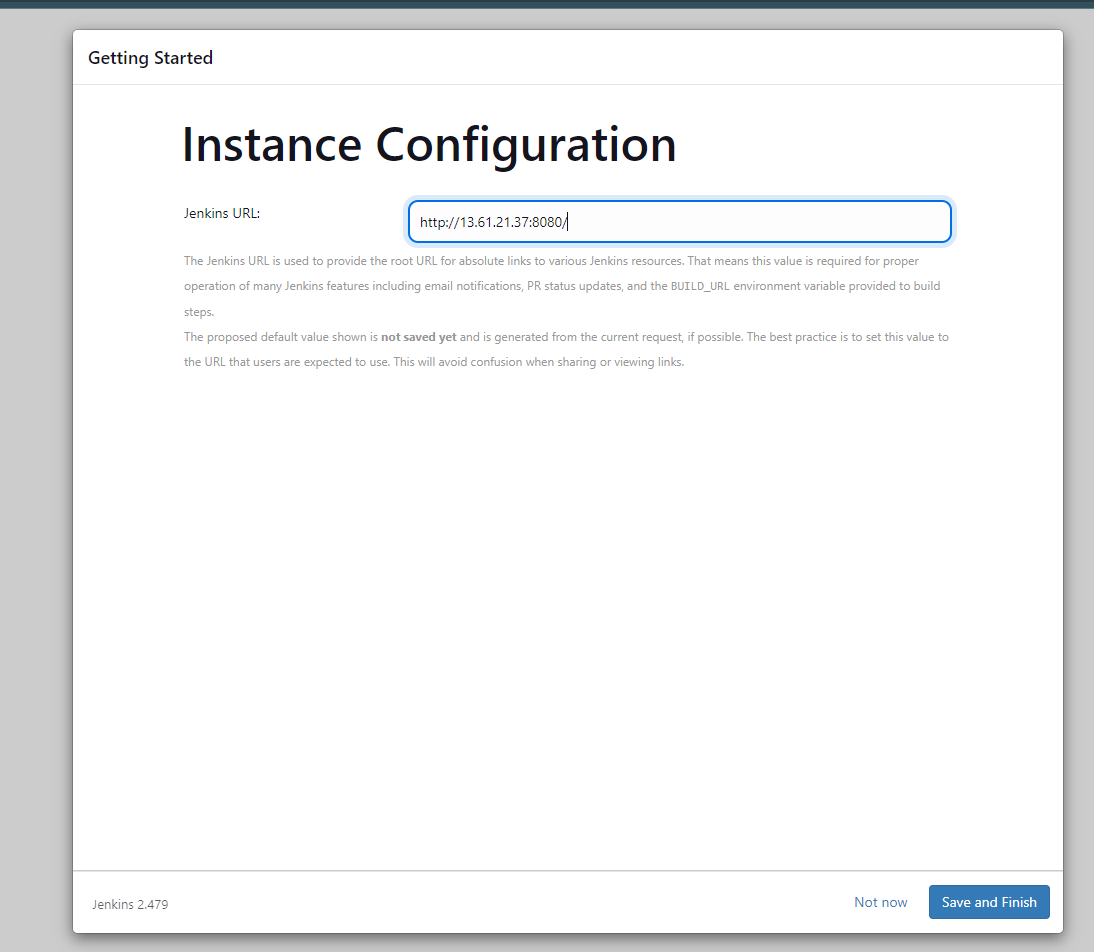


v. Once done, enter the Admin User details.

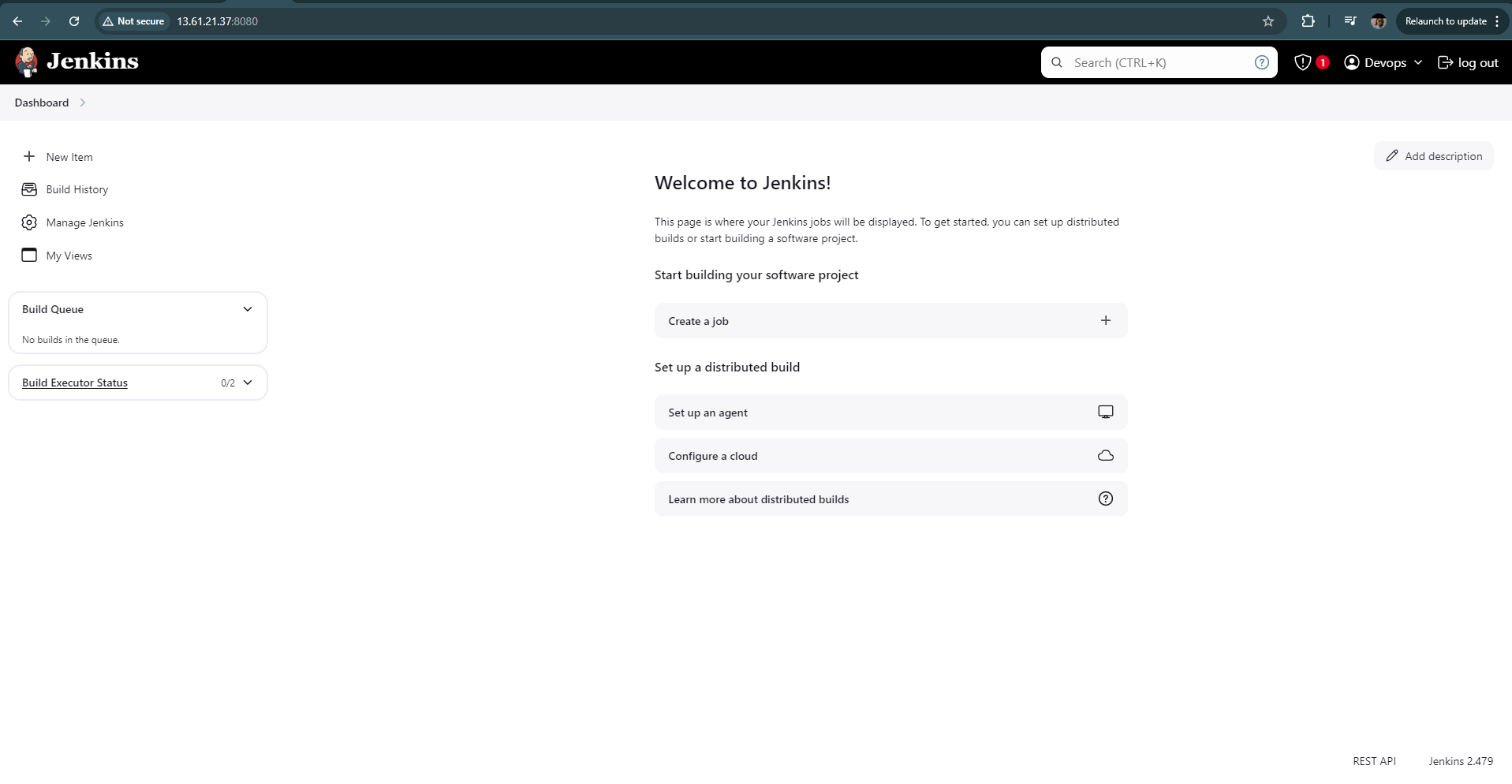
I have created a account with the name devops and mail id:patelsanvi596@gmail.com but certainly I have clicked next so I logged out and log in again



Then click on Save and Continue.



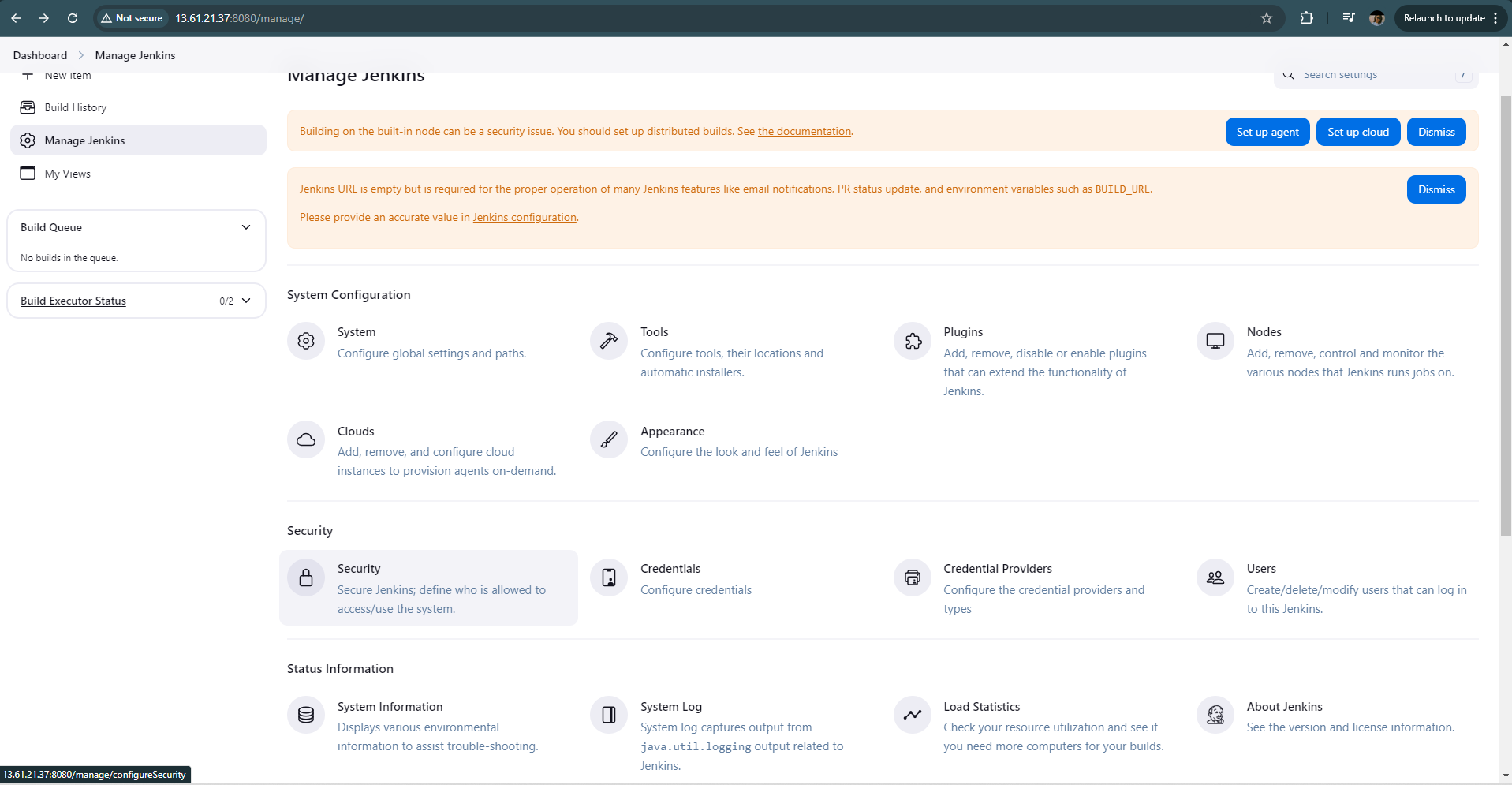
Again, click Save and Finish. Click on Install Suggested Plugins. Once it's done, we will land on a page as shown below.



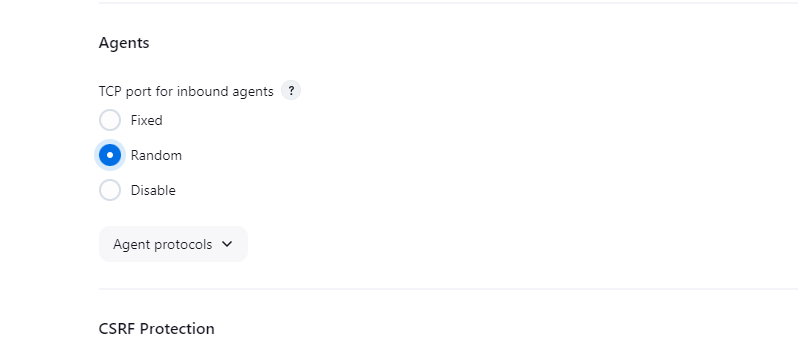
This is our Jenkins Dashboard.

***b). Configure Slave nodes***

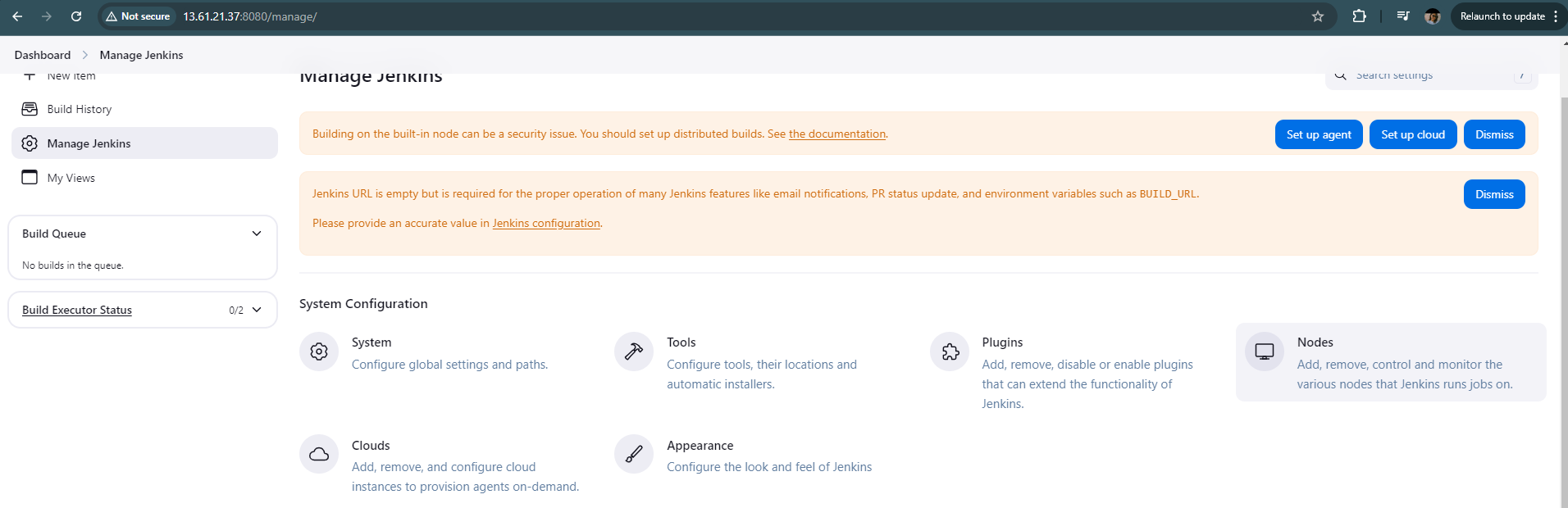
i. Go to Manage Jenkins. Click on Configure Global Security.



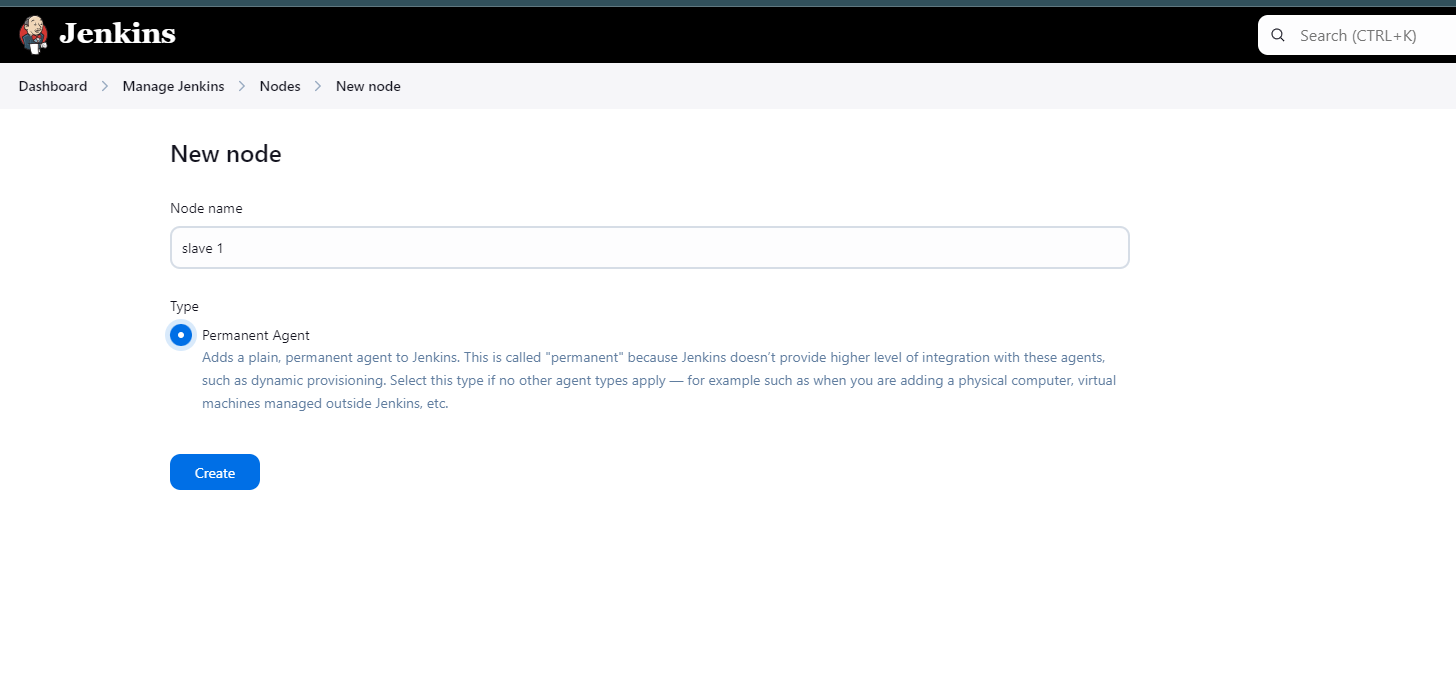
ii. Change the Agents to Random. Then click on Save.



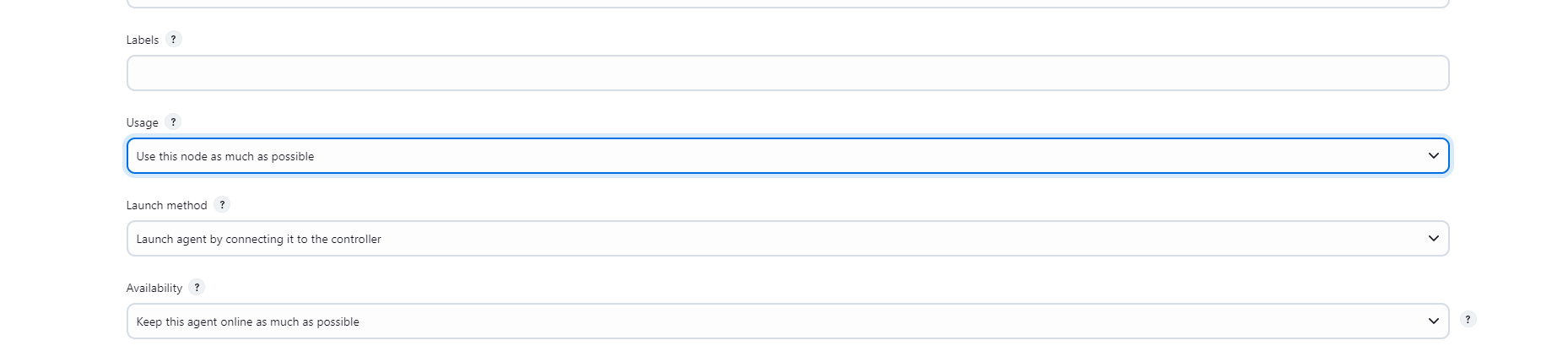
iii. Now go to Manage Nodes.



iv. Click on New Node. Add Slave1 as new node and make Permanent Agent. Click on ok.

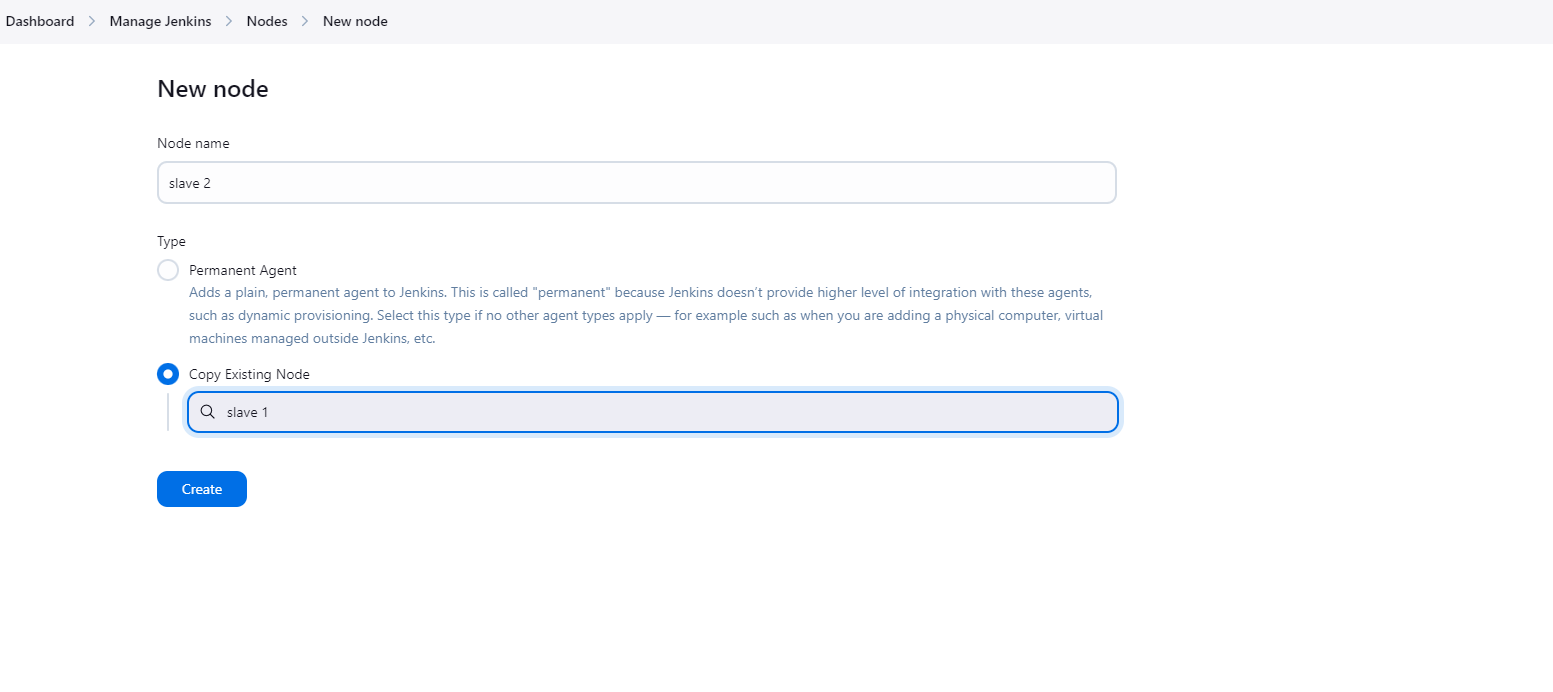


v. Go to Launch method change it to **Launch agent by connecting it to the controller**.

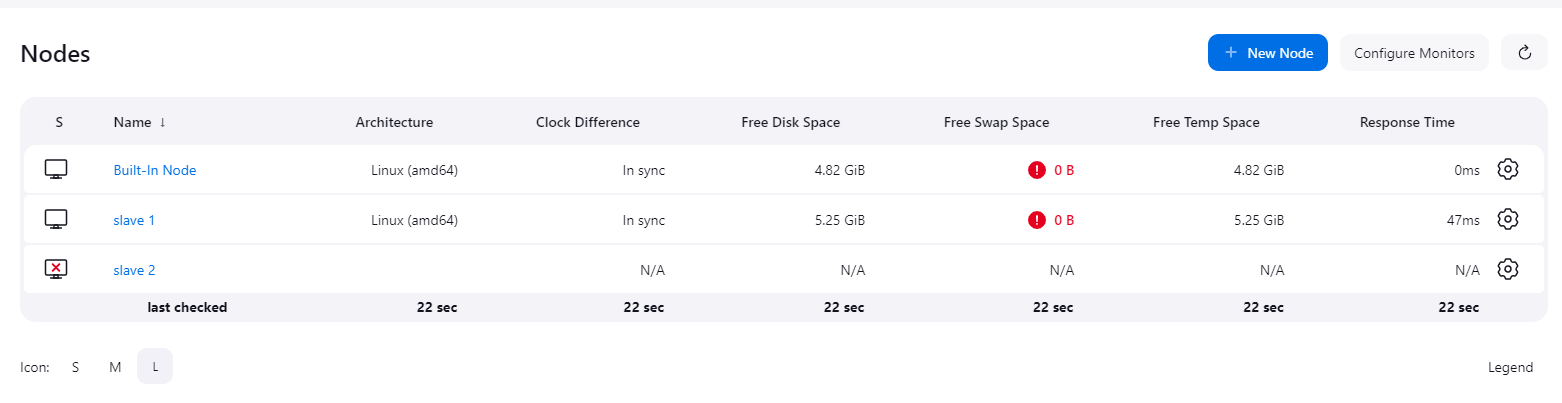


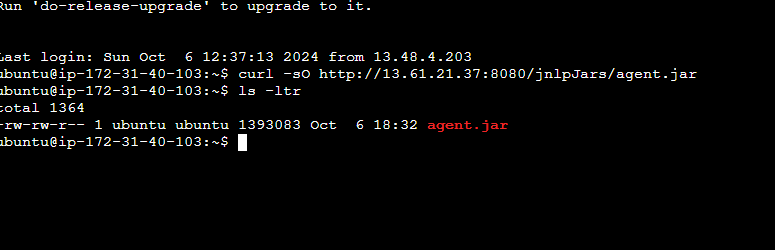
vi. Then add the current working directory path to **/home/ubuntu/jenkins**. Then click on Save.



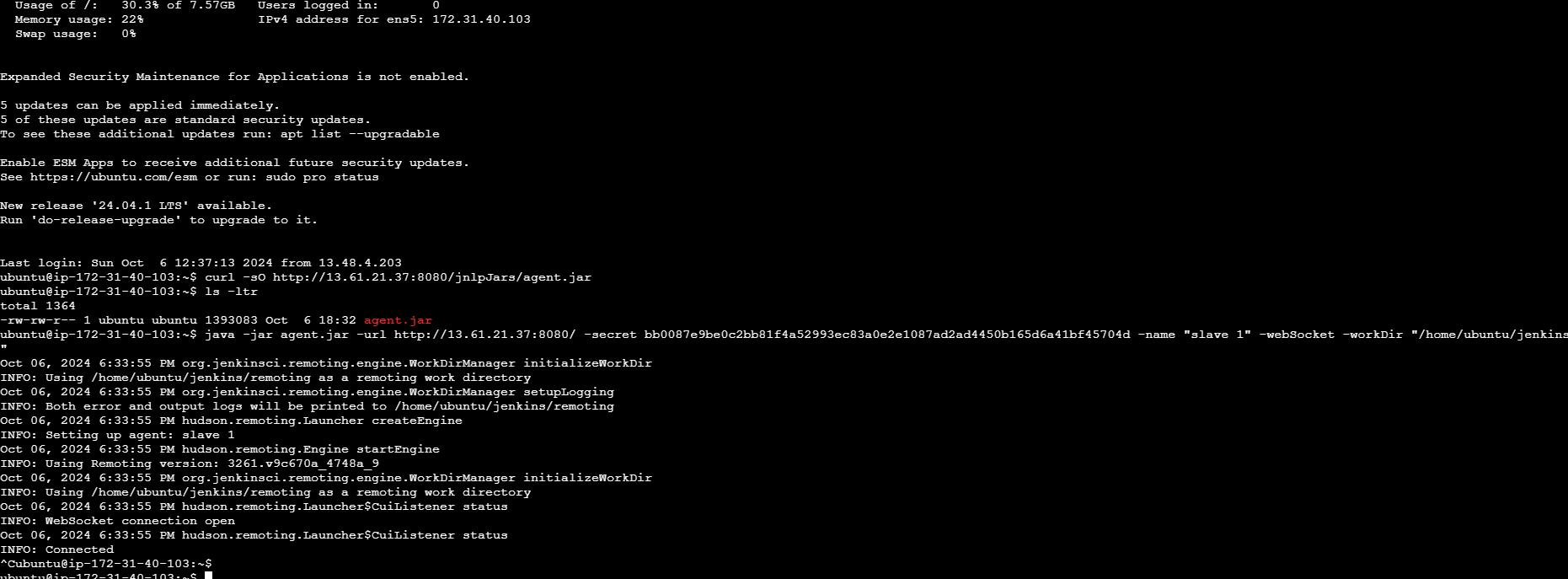


vii. Then click ok. You can see the list of nodes that we have on the Jenkins Dashboard.

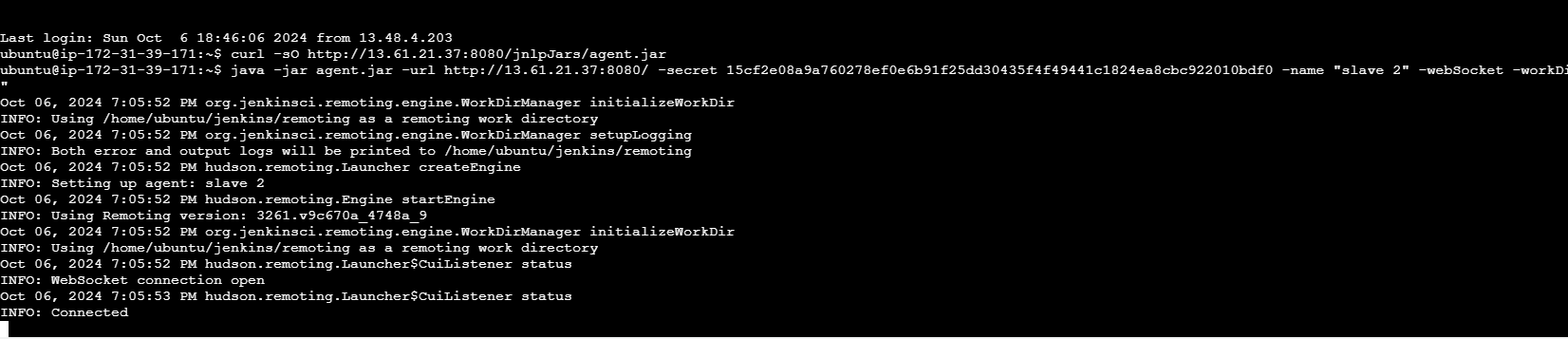




Slave 1 is successfully connected.



**SLAVE 2**

****

1. **Create a Jenkins job to clone repo**[***https://github.com/vistasunil/devopsIQ***](https://github.com/vistasunil/devopsIQ)**and deploy the website inside it the slave instance in container.**

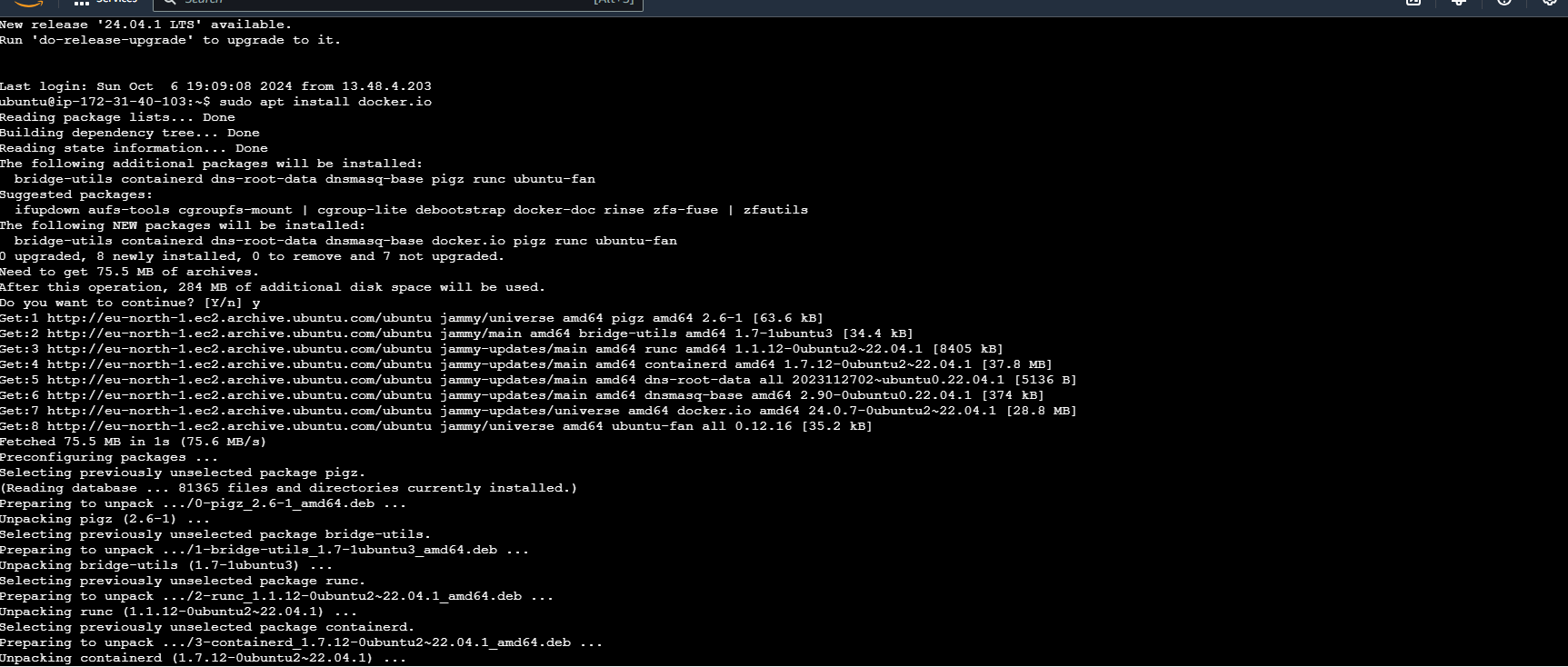
***Solution:***

**i. Open your GitHub account and import the below given repository.**

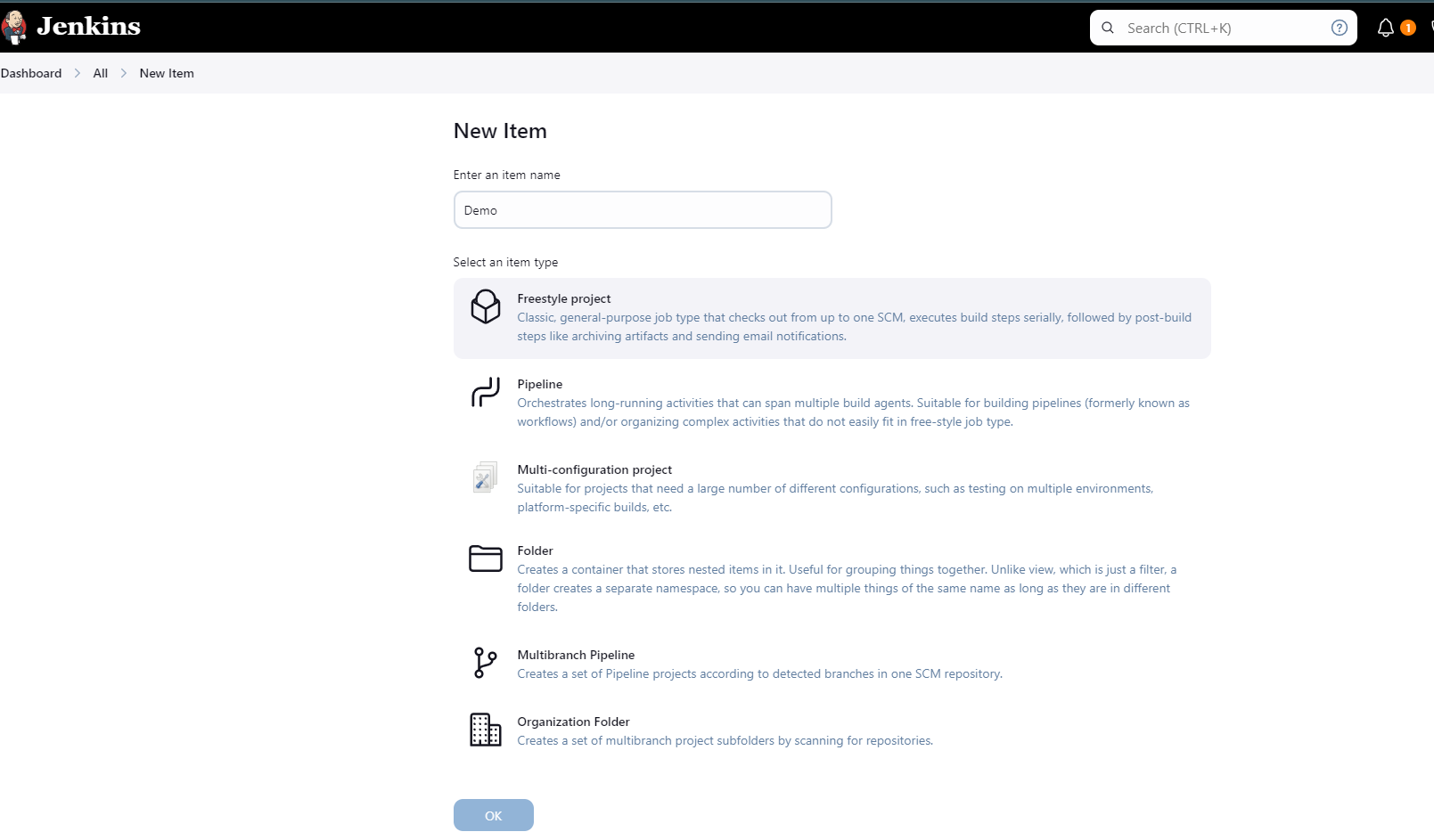
[***https://github.com/vistasunil/devopsIQ***](https://github.com/vistasunil/devopsIQ)

**ii. Install docker on both Slave1**

**sudo apt install docker.io**

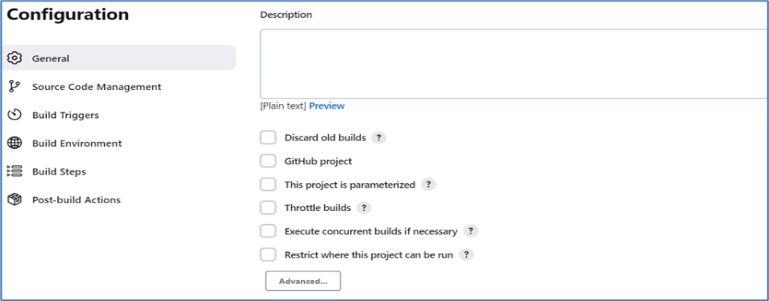
****

**iii. Open Jenkins Dashboard. Create a new job (Freestyle Project) for Slave1**

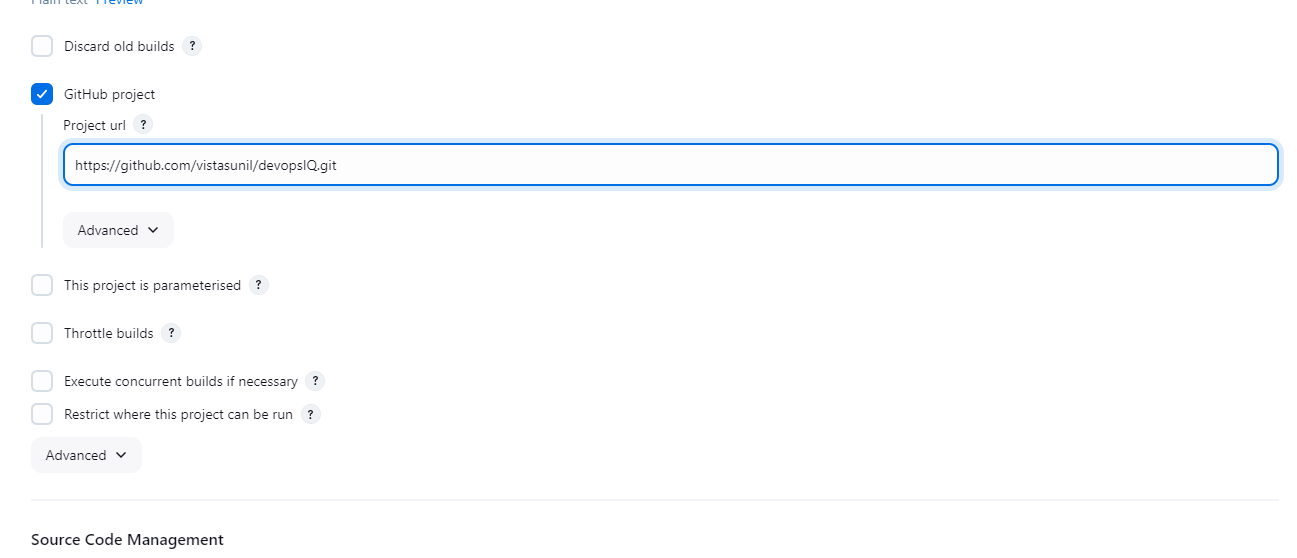
****

**Then click on Ok.**

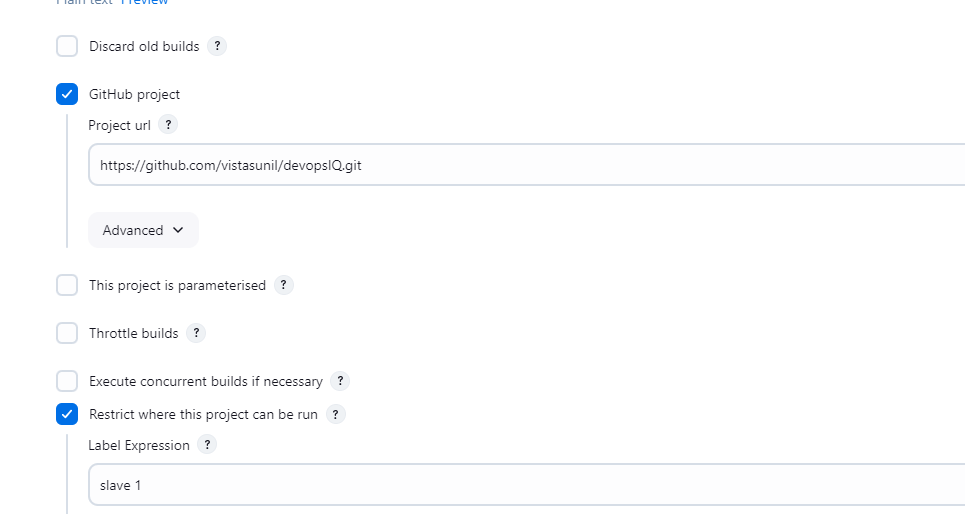
**You should land on a page like this.**

****

v. Place your git repository link as shown below.



vi. Click on Restrict where this project can be run. Add Slave1 there.

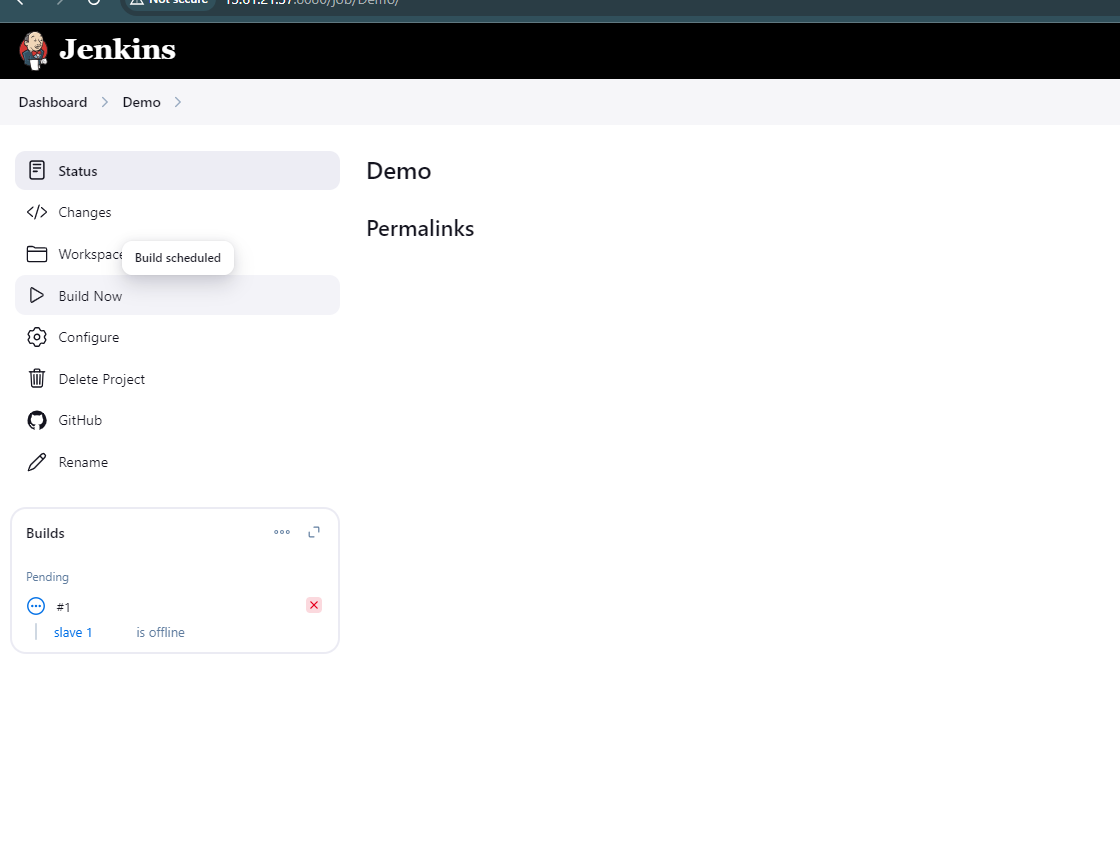


vii. Go to Source Code Management, click on git, add the git repository link there as well.

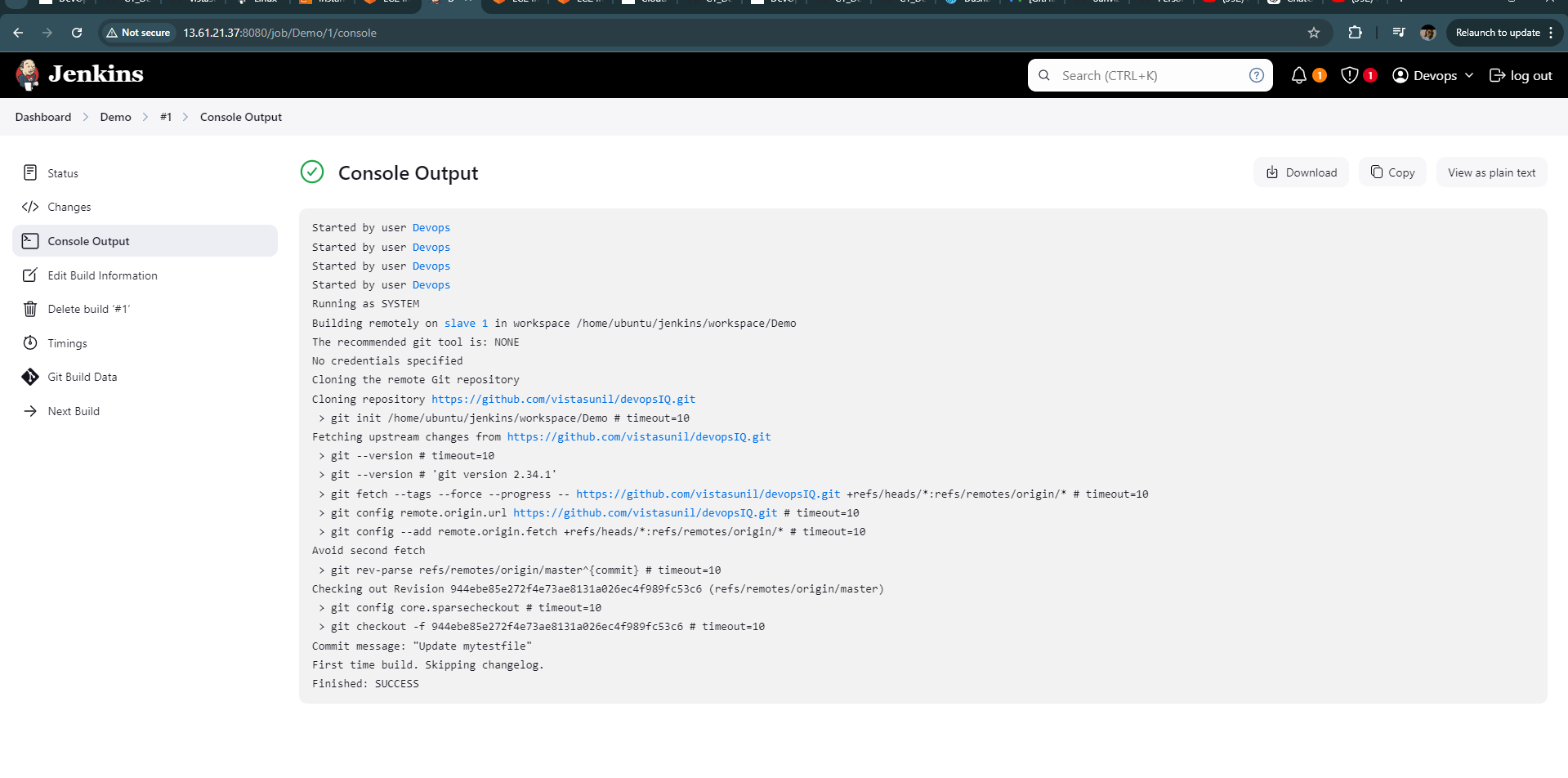


Click on Save.

viii. Click on Build Now, if the building is done without any error there will be blue circle in the building history.



ix. Click on the blue circle of build #1.



You can see it has been built successfully. Let us verify that.

x. Go to slave1.

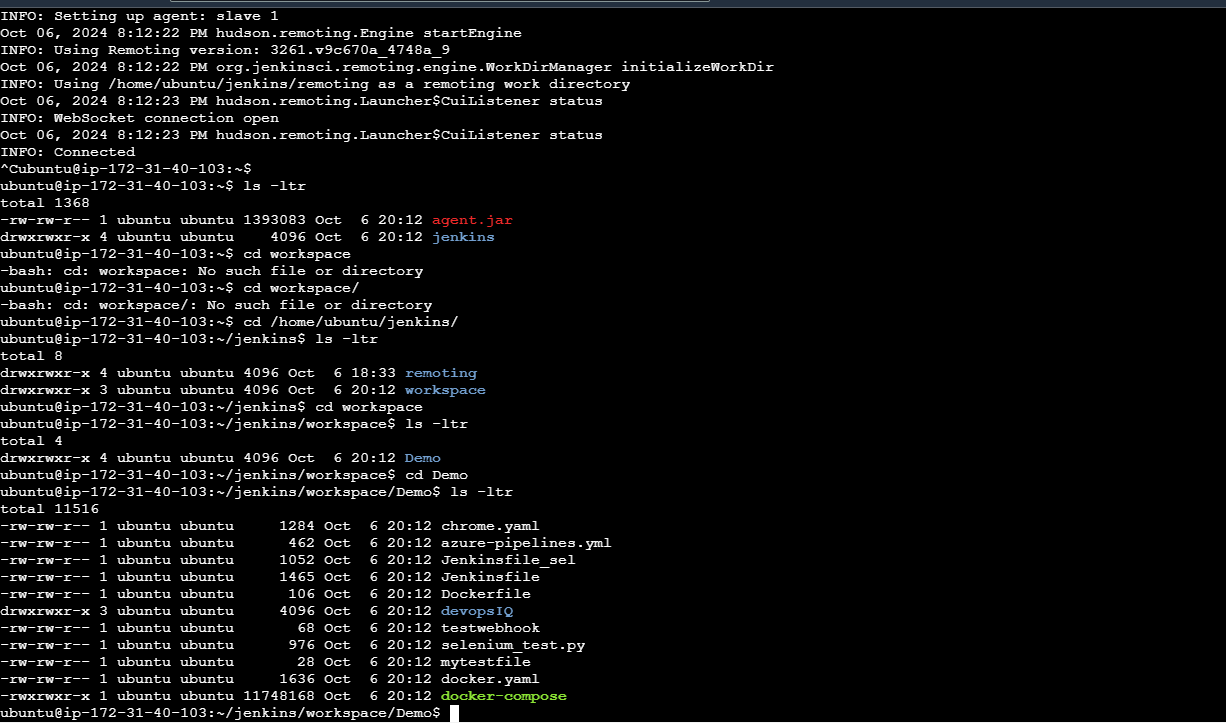
ls -ltr

cd workspace

ls -ltr

cd Demo

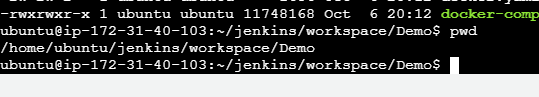
ls -ltr



You can see the repository files there. This means the git repository has been successfully cloned into the Demo job.

**Now we will deploy the website that we have stored in our repository.**

xi. To run the Dockerfile we have to check the copy the present working directory.

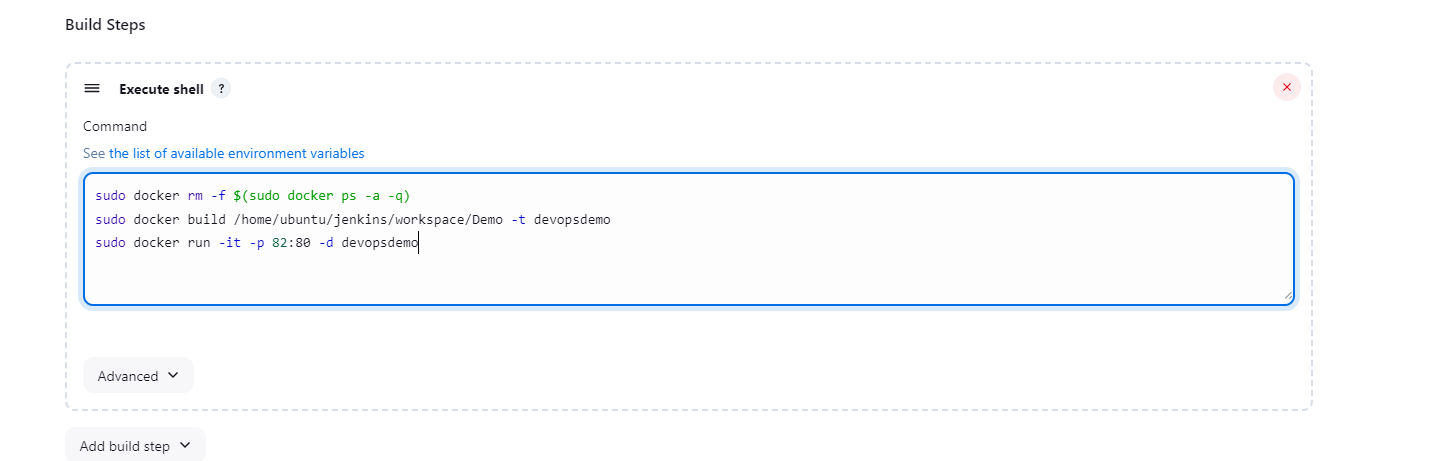


xii. Now go back to configuring the job. Click on Build, then go to Execute shell

sudo docker rm -f $(sudo docker ps -a -q)

sudo docker build /home/ubuntu/jenkins/workspace/Demo -t devopsdemo

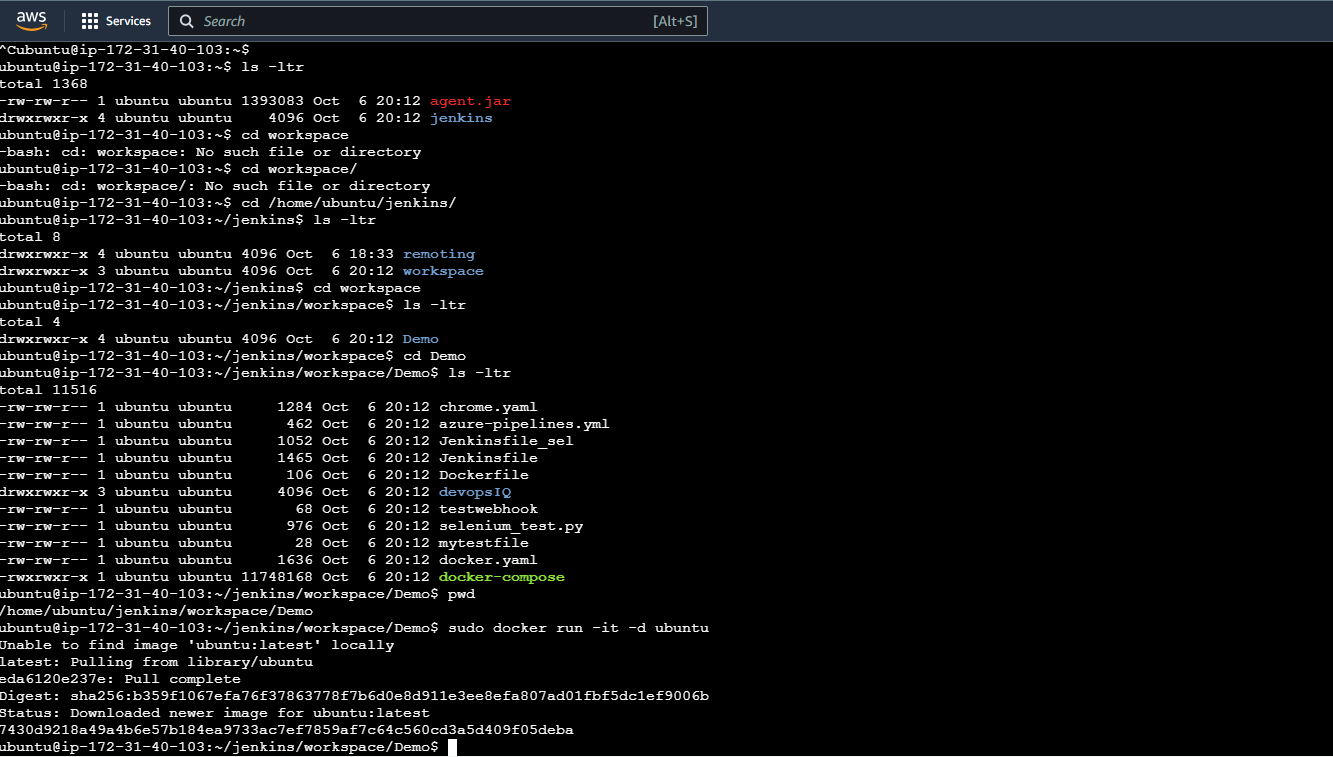
sudo docker run -it -p 82:80 -d devopsdemo



Click on save.

xiii. Before building our job again we must add one arbitrary container in slave1. Add container by performing the following command.

sudo docker run -it -d ubuntu



Now we have added a container as below

